

- [106] B. Cox, D. Evans, A. Filipi, J. Rowanhill, W. Hu, J. Davidson, J. Knight, A. Nguyen-Tuong, and J. Hiser, “N-variant systems: a secretless framework for security through diversity,” in *Proc. of USENIX Security Symposium*, USENIX-SS’06, 2006.
- [107] D. Bruschi, L. Cavallaro, and A. Lanzi, “Diversified process replicas for defeating memory error exploits,” in *Proc. of the Int. Performance, Computing, and Communications Conference*, 2007.
- [108] B. Salamat, A. Gal, T. Jackson, K. Manivannan, G. Wagner, and M. Franz, “Stopping buffer overflow attacks at run-time: Simultaneous multi-variant program execution on a multicore processor,” tech. rep., Technical Report 07-13, School of Information and Computer Sciences, UC Irvine, 2007.
- [109] L. Davi, C. Liebchen, A.-R. Sadeghi, K. Z. Snow, and F. Monrose, “Isomeron: Code randomization resilient to (just-in-time) return-oriented programming,” in *NDSS*, 2015.
- [110] G. Agosta, A. Barengi, G. Pelosi, and M. Scandale, “The MEET approach: Securing cryptographic embedded software against side channel attacks,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 8, pp. 1320–1333, 2015.
- [111] T. Jackson, B. Salamat, A. Homescu, K. Manivannan, G. Wagner, A. Gal, S. Brunthaler, C. Wimmer, and M. Franz, “Compiler-generated software diversity,” in *Moving Target Defense*, pp. 77–98, Springer, 2011.
- [112] A. Amarilli, S. Müller, D. Naccache, D. Page, P. Rauzy, and M. Tunstall, “Can code polymorphism limit information leakage?,” in *IFIP International Workshop on Information Security Theory and Practices*, pp. 1–21, Springer, 2011.
- [113] A. Voulimeneas, D. Song, P. Larsen, M. Franz, and S. Volckaert, “dmvx: Secure and efficient multi-variant execution in a distributed setting,” in *Proceedings of the 14th European Workshop on Systems Security*, pp. 41–47, 2021.
- [114] R. Tsoupidi, R. C. Lozano, and B. Baudry, “Constraint-based diversification of JOP gadgets,” *CoRR*, vol. abs/2111.09934, 2021.
- [115] J.-R. Falleri, F. Morandat, X. Blanc, M. Martinez, and M. Monperrus, “Fine-grained and accurate source code differencing,” in *Proceedings of the International Conference on Automated Software Engineering*, pp. 313–324, 2014.
- [116] H. Bostani and V. Moonsamy, “Evadedroid: A practical evasion attack on machine learning for black-box android malware detection,” *CoRR*, vol. abs/2110.03301, 2021.

- [117] D. Yao, X. Shu, L. Cheng, S. J. Stolfo, E. Bertino, and R. Sandhu, *Anomaly detection as a service: challenges, advances, and opportunities*. Springer, 2018.
- [118] S. A. Hofmeyr, S. Forrest, and A. Somayaji, “Intrusion detection using sequences of system calls,” *J. Comput. Secur.*, vol. 6, p. 151–180, aug 1998.
- [119] J. Cabrera Arteaga, M. Monperrus, and B. Baudry, “Scalable comparison of javascript v8 bytecode traces,” in *Proceedings of the 11th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages, VMIL 2019, (New York, NY, USA)*, p. 22–31, Association for Computing Machinery, 2019.
- [120] Y. Fang, C. Huang, L. Liu, and M. Xue, “Research on malicious javascript detection technology based on lstm,” *IEEE Access*, vol. 6, pp. 59118–59125, 2018.
- [121] E. Johnson, D. Thien, Y. Alhessi, S. Narayan, F. Brown, S. Lerner, T. McMullen, S. Savage, and D. Stefan, “, : Sfi safety for native-compiled wasm,” *Network and Distributed Systems Security (NDSS) Symposium*, 2021.
- [122] C. Fred, “Computer viruses,” in *Proceedings of the 7th DoD/NBS Computer Security Conference 1984*, pp. 240–263, 1986.
- [123] R. L. Castro, C. Schmitt, and G. D. Rodosek, “Armed: How automatic malware modifications can evade static detection?,” in *2019 5th International Conference on Information Management (ICIM)*, pp. 20–27, 2019.
- [124] R. L. Castro, C. Schmitt, and G. Dreo, “Aimed: Evolving malware with genetic programming to evade detection,” in *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 240–247, IEEE, 2019.
- [125] H. Aghakhani, F. Gritti, F. Mecca, M. Lindorfer, S. Ortolani, D. Balzarotti, G. Vigna, and C. Kruegel, “When malware is packin’ heat: limits of machine learning classifiers based on static analysis features,” in *Proc. of NDSS*, 2020.
- [126] M. Chua and V. Balachandran, “Effectiveness of android obfuscation on evading anti-malware,” in *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy, CODASPY ’18*, Association for Computing Machinery, 2018.
- [127] P. Dasgupta and Z. Osman, “A Comparison of State-of-the-Art Techniques for Generating Adversarial Malware Binaries,” *arXiv e-prints*, Nov. 2021.